School of Computation, Information and Technology
Technische Universität München

Lehrstuhl für
Rechnerarchitektur & Parallele Systeme
Prof. Dr. Martin Schulz
Dominic Prinz
Jakob Schäffeler

Lehrstuhl für
Design Automation
Prof. Dr.-Ing. Robert Wille
Stefan Engels

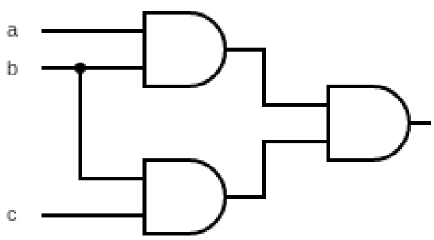
Einführung in die Rechnerarchitektur

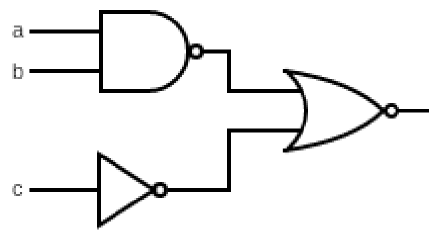
Wintersemester 2024/2025

Übungsblatt 13: SAT und Physical Design 27.01.2025 – 31.01.2025

1 Verifikation

Betrachte folgende Schaltkreise:





a) Bilde den Miterschaltkreis für die beiden Schaltungen.

b) Überführe den Miter mittels Tseitin-Transformation in eine CNF.

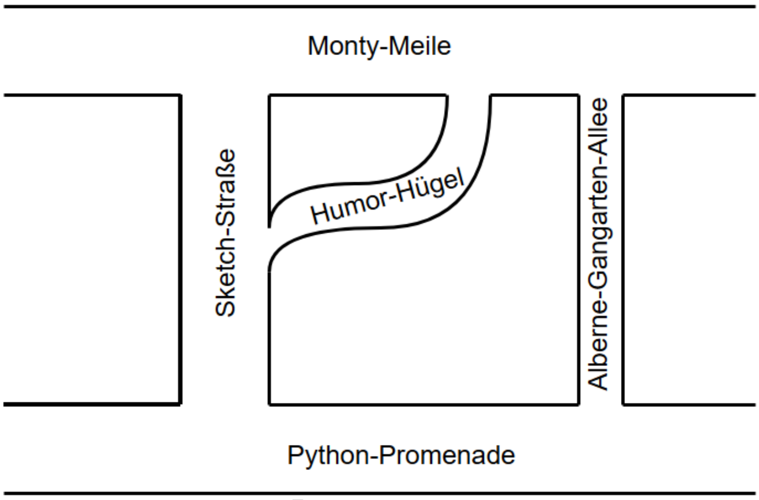
© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

c) Löse die o.g. Formulierung mittels eines SAT-Solvers. Sind die Schaltkreise äquivalent?

Hinweis: Man kan die o.g. Formel z.B. im DIMACS CNF Format[1] *angeben und dann online auf* ***https://jgalenson.github.io/research.js/demos/minisat.html*** *lösen lassen.*

2 Ministerium für Alberne Gangarten

a) Das Ministerium für Alberne Gangarten [2] hat in letzter Zeit von der Regierung weniger Haushaltsmittel zur Verfügung gestellt bekommen als für die nationale Sicherheit. Damit ist das Ministerium chronisch unterfinanziert und kann nur wenige alberne Gänge staatlich fördern, weshalb Großbritannien droht international abgehängt zu werden. Mr. Teabag, ziviler Mitarbeiter, plant daher eine Werbeoffensive. Hierzu sollen die Straßen eines Dorfes nahe London mit entsprechenden Graffiti verziert werden, um die Aufmerksamkeit der Zivilbevölkerung zu gewinnen. Als Motive sollen Silhouetten des 12-teiligen albernen Gangs aus Abbildung 2 verwendet werden. Das Dorf besteht aus insgesamt fünf Straßen:



Auf Grund des knappen Budgets, können jedoch maximal drei Motive in Auftrag gegeben werden, nämlich 1 , 2  und 3 . Um dennoch eine gewisse Diversität im Straßenbild zu gewährleisten hat das Ministerium folgende Bedingungen aufgestellt:

i) Innerhalb einer Straße wird ausschließlich eines der Motive verwendet.

ii) Zwei aneinandergrenzende Straßen müssen zwingend mit unterschiedlichen Motiven verziert sein.

Unterstüze Mr. Teabag dabei die drei Motive auf die fünf Straßen zu verteilen unter Beachtung der vom Ministerium aufgestellten Bedingungen.

- Formuliere das Problem als ein Graphenfärbenproblem.

[1] https://people.sc.fsu.edu/~jburkardt/data/cnf/cnf.html
[2] https://www.youtube.com/watch?v=iV2ViNJFZC8

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

Lösungsvorschlag

Wir bezeichnen die drei „Farben“ mit  =1,  =2 und  =3. Hierdurch ergeben sich insgesamt 15 binäre Variablen:

$$\left\{x_{(v,f)} \in \{0,1\} \colon v \in \{M,H,S,A,P\}, f \in \{1,2,3\}\right\}$$

Da jeder Knoten gefärbt sein muss, ergeben sich außerdem die fünf Nebenbedingungen:

$$x_{(v,1)} \vee x_{(v,2)} \vee x_{(v,3)} \qquad \forall v \in \{M,H,S,A,P\}$$

Für jede Kante $\{u,v\} \in E := \{\{M,H\},\{M,S\},\{M,A\},\{S,H\},\{S,P\},\{A,P\}\}$ müssen u und v verschiedene Farben haben:

$$\neg x_{(u,f)} \vee \neg x_{(v,f)} \qquad \forall f \in \{1,2,3\}$$

Durch Verundung ergibt sich die KNF mit insgesamt $5 + 6 \cdot 3 = 23$ Klauseln:

$$\left(\bigwedge_{v \in \{M,H,S,A,P\}} \left(x_{(v,1)} \vee x_{(v,2)} \vee x_{(v,3)}\right)\right) \wedge \left(\bigwedge_{\{u,v\} \in E} \bigwedge_{f \in \{1,2,3\}} \left(\neg x_{(u,f)} \vee \neg x_{(v,f)}\right)\right)$$

insgesamt, also:

$$\begin{array}{ll}
(x_{(M,1)} \vee x_{(M,2)} \vee x_{(M,3)}) & \wedge \\
(x_{(H,1)} \vee x_{(H,2)} \vee x_{(H,3)}) & \wedge \\
(x_{(S,1)} \vee x_{(S,2)} \vee x_{(S,3)}) & \wedge \\
(x_{(A,1)} \vee x_{(A,2)} \vee x_{(A,3)}) & \wedge \\
(x_{(P,1)} \vee x_{(P,2)} \vee x_{(P,3)}) & \wedge \\
(\neg x_{(M,1)} \vee \neg x_{(H,1)}) & \wedge \\
(\neg x_{(M,2)} \vee \neg x_{(H,2)}) & \wedge \\
(\neg x_{(M,3)} \vee \neg x_{(H,3)}) & \wedge \\
(\neg x_{(M,1)} \vee \neg x_{(S,1)}) & \wedge \\
(\neg x_{(M,2)} \vee \neg x_{(S,2)}) & \wedge \\
(\neg x_{(M,3)} \vee \neg x_{(S,3)}) & \wedge \\
(\neg x_{(M,1)} \vee \neg x_{(A,1)}) & \wedge \\
(\neg x_{(M,2)} \vee \neg x_{(A,2)}) & \wedge \\
(\neg x_{(M,3)} \vee \neg x_{(A,3)}) & \wedge \\
(\neg x_{(H,1)} \vee \neg x_{(S,1)}) & \wedge \\
(\neg x_{(H,2)} \vee \neg x_{(S,2)}) & \wedge \\
(\neg x_{(H,3)} \vee \neg x_{(S,3)}) & \wedge \\
(\neg x_{(S,1)} \vee \neg x_{(P,1)}) & \wedge \\
(\neg x_{(S,2)} \vee \neg x_{(P,2)}) & \wedge \\
(\neg x_{(S,3)} \vee \neg x_{(P,3)}) & \wedge
\end{array}$$

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

$$
\begin{array}{ll}
(\neg x_{(A,1)} \vee \neg x_{(P,1)}) & \wedge \\
(\neg x_{(A,2)} \vee \neg x_{(P,2)}) & \wedge \\
(\neg x_{(A,3)} \vee \neg x_{(P,3)}) &
\end{array}
$$

- Nutze einen SAT-Solver (z.B. `https://jgalenson.github.io/research.js/demos/minisat.html`) um eine valide Aufteilung von Motiven zu Straßen zu generieren.

Lösungsvorschlag

Mit der Variablenordnung $x_{(M,1)}$, $x_{(M,2)}$, $x_{(M,3)}$, $x_{(H,1)}$, $x_{(H,2)}$, $x_{(H,3)}$, $x_{(S,1)}$, $x_{(S,2)}$, $x_{(S,3)}$, $x_{(A,1)}$, $x_{(A,2)}$, $x_{(A,3)}$, $x_{(P,1)}$, $x_{(P,2)}$, $x_{(P,3)}$ ergibt sich folgende CNF-Datei:

```
c Blatt 13 Aufgabe 2a
c
p cnf 15 23
1 2 3 0
4 5 6 0
7 8 9 0
10 11 12 0
13 14 15 0
-1 -4 0
-2 -5 0
-3 -6 0
-1 -7 0
-2 -8 0
-3 -9 0
-1 -10 0
-2 -11 0
-3 -12 0
-4 -7 0
-5 -8 0
-6 -9 0
-7 -13 0
-8 -14 0
-9 -15 0
-10 -13 0
-11 -14 0
-12 -15 0
```

Als Solver-Output könnten wir z.B. folgendes erhalten:

```
CPU time: 0.001s
SAT 1 -2 -3 -4 5 -6 -7 -8 9 -10 -11 12 13 -14 -15
```

d.h. die Variablen mit Nummern 1, 5, 9, 12 und 13 sind jeweils wahr. Dies enstpricht nach der o.g. Reihenfolge $x_{(M,1)}$, $x_{(H,2)}$, $x_{(S,3)}$, $x_{(A,3)}$ und $x_{(P,1)}$.

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

- Formuliere das Problem davon ausgehend als SAT-Instanz in KNF.

- Nutze einen SAT-Solver (z.B. `https://jgalenson.github.io/research.js/demos/minisat.html`) um eine valide Aufteilung von Motiven zu Straßen zu generieren.

b) Betrachte das folgende Programm mit fünf Variablen:

```
s1: t1 = 5;
s2: t2 = 6;
s3: t3 = 7;
s4: t4 = t1 + t3;
s5: t5 = t2 + 3;
s6: t3 = t4 + t5;
s7: t1 = t3 / t5;
```

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme alle Rechte vorbehalten

- Erzeuge den Registerkonfliktgraphen.

	s1	s2	s3	s4	s5	s6	s7
t1							
t2							
t3							
t4							
t5							

- Finde eine konfliktfreie Verteilung der Variablen auf maximal drei Register.

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

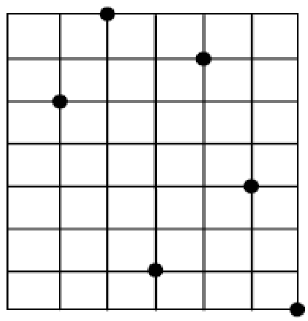
Hinweis: Nutze Teilaufgabe a

c) *Bonus:* Der Minister befürchtet, dass die Werbekampagne nicht den gewünschten Effekt erzielt, wenn er nicht bald auch zusätzlich bereits neue alberne Gänge präsentieren kann. Unterstütze den Minister und entwickel einen neuen albernen Gang.

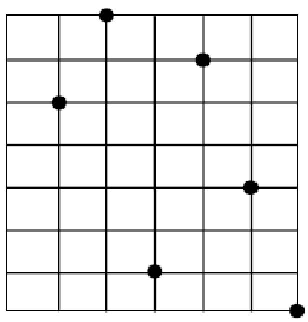
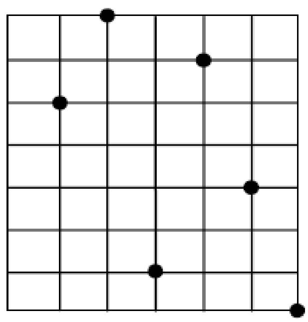
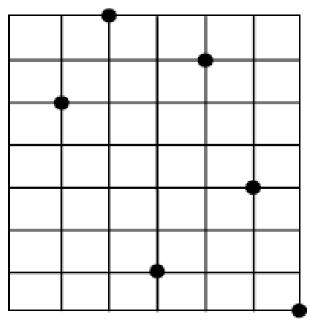
Hinweis: Wir beschränken uns an dieser Stelle (im Gegensatz zu Mr. Teabag) auf alberne Gänge, die das pünktliche Erscheinen (z.B. zu ERA-Veranstaltungen) nicht gefährden.

3 Single-Net Routing

Gegeben seien sechs Pins eines Netzes auf einem horizontalen und vertikalen Raster, wobei jede Kante des Katters mit Kosten 1 belegt sei:

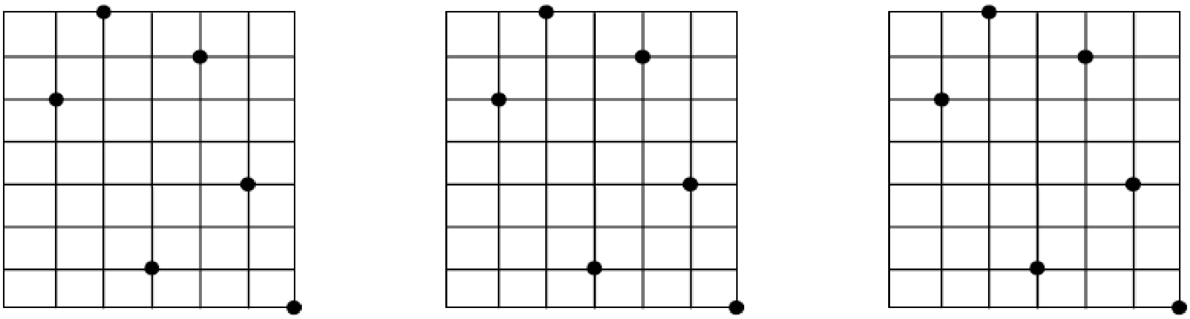


a) Stelle graphisch sämtliche Hanan-Punkte dar.

b) Erzeuge einen (möglichst minimalen) rektilinearen Steinerbaum unter Nutzung der in der Vorlesung vorgestellten Heuristik.

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

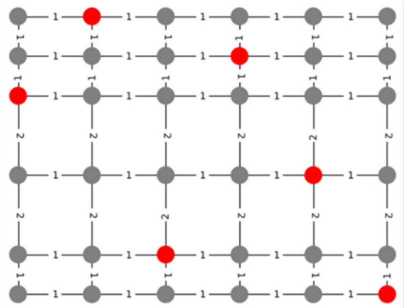


c) Gebe die Anzahl der Steinerpunkte des Baumes und deren jeweiligen Knotengrad an.

d) Formuliere das Mixed Integer Linear Program (MILP), um das o.g. Problem exakt zu lösen.

Hinweis: Es ist nicht notwendig, jede Nebenbedingung explizit aufzuschreiben. Alle möglicherweise verwendeten Mengen und Konstanten sollten jedoch explizit definiert werden. Auch graphische Definitionen zählen in diesem Kontext als „explizit“.

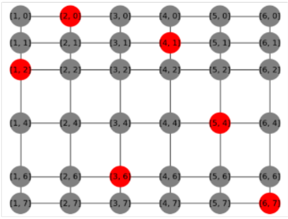
© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme alle Rechte vorbehalten



e) Wie viele Nebenbedingungen hat das in Teilaufgabe d formulierte MILP.

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

f) Schreibe eine beliebige Nebenbedingungen explizit aus.



g) Nach kurzer Zeit wird der Prozess des MILP-Solvers unterbrochen bevor eine Lösung zurückgegeben werden konnte. Der bis dahin erkundete Branch&Bound-Baum des Solvers ist in Abbildung 3 schematisch abgebildet.

Ist die in Teilaufgabe b berrechnete heuristische Lösung optimal? Begründe die Antwort. Mögliche Antworten sind:

- ☐ Die Lösung aus Teilaufgabe b ist optimal.
- ☐ Die Lösung aus Teilaufgabe b ist nicht optimal.
- ☐ Die gegebenen Informationen reichen nicht aus, um die Optimalität zu beweisen oder zu widerlegen.

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme alle Rechte vorbehalten

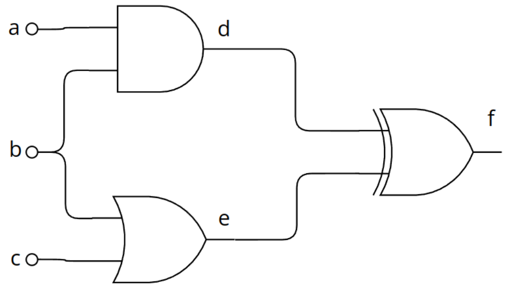
4 Online Ressources

Versuche ein Gefühl für optimale Steinerbäume zu entwickeln. Spiele hierzu z.B. ein paar Runden des Spiels „Routing“ -> „Steinerbäume“ unter `https://www.arithmeum.uni-bonn.de/ausstellungen/chipausstellung/videos-spiele-zum-chipdesign.html`

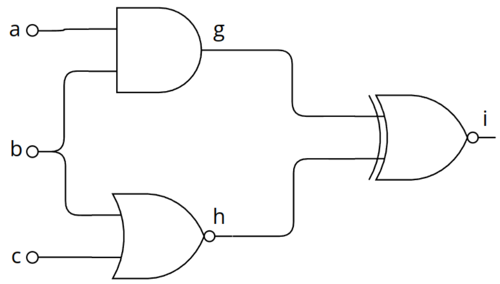
5 Verifikation mit SAT (Hausaufgabe)

Bearbeitung und Abgabe der Hausaufgabe 13 auf `https://artemis.in.tum.de/courses/401` bis **Sonntag, den 02.02.2025, 23:59 Uhr**.

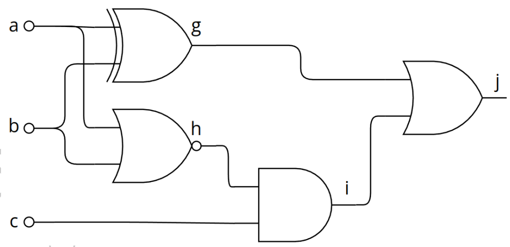
Gegeben ist folgende Augangsschaltung:



Im Folgenden wird die Äquivalenz dieser beiden Vergleichsschätzungen mithilfe von SAT überprüft:



(a) Vergleichsschaltung I



(b) Vergleichsschaltung II

a) Überprüfe die Äquivalenz der Ausgansschaltung zu den beiden Vergleichsschaltungen. Folge dabei wieder den Schritten, wie in Aufgabe 1:

- Erstelle die zwei Miter bestehend aus der ersten Schaltung und der jeweiligen Vergleichsschaltung.
- Ermittle mithilfe der Tseitin-Transformation die CNF der beiden Miter.
- Löse anschließend die o.g. Formulierung mittels des SAT-Solvers auf `https://jgalenson.github.io/research.js/demos/minisat.html`.

Hinweis: Verwende für die Abgabe das DIMACS CNF Format[3]*. Die Variablen in diesem Format sind Integer. Um eine konsistente Namensgebung zu gewährleisten, sollten die Variablen bei 1 beginnen und alphabetisch aufsteigend nummeriert werden. Entsprechend gilt: $a = 1$, $b = 2$, $c = 3$ usw.*

Erstelle für jeden Miter eine Datei, die die zugehörigen SAT-Klauseln enthält. Gib außerdem an, ob die entsprechenden Netzwerke äquivalent sind oder nicht. Falls eines der Netzwerke nicht äquivalent sein sollte, gebe zusätzlich eine Eingangsbelegung an, bei der die verglichenen Schaltkreise verschiedene Ausgaben liefern.

b) Eine der beiden Vergleichsschaltungen ist nicht äquivalent zur Ausgangsschaltung. Uns fällt auf, dass die vom SAT-Solver ausgegebene Belegung der Inputs für uns eigentlich ein don't care sein sollte. Füge *eine* SAT Klausel hinzu, die diesen Input ausschließt. Gib an ob nun die Schaltungen äquivalent sind oder nicht.

[3] `https://people.sc.fsu.edu/~jburkardt/data/cnf/cnf.html`

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

6 Referenzmaterial

Abbildung 2: Beispiel eines albernen Gangs. Bild: Jazeen Hollings/Wikimedia Commons

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

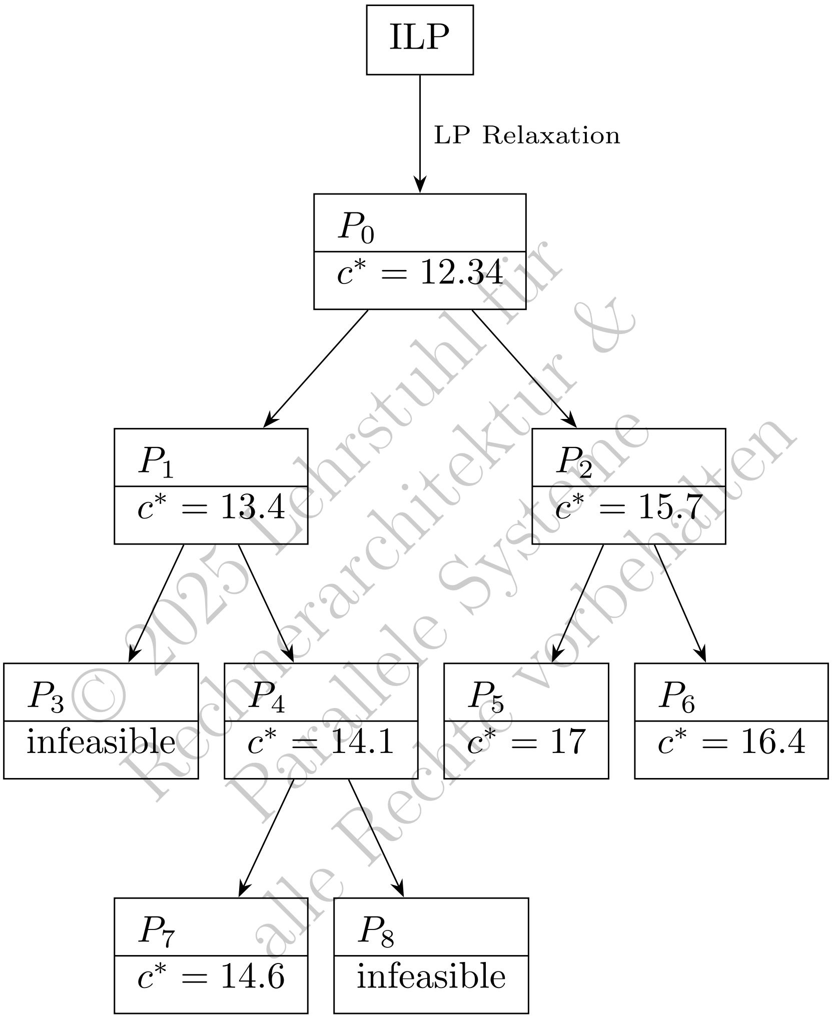


Abbildung 3: Branch&Bound-Baum

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten

© 2025 Lehrstuhl für Rechnerarchitektur & Parallele Systeme, alle Rechte vorbehalten